基于改进 PVANet 的实时小目标检测方法 *

段秉环†，文鹏程，李　鹏

(中国航空工业集团公司西安航空计算技术研究所 机载弹载计算机航空科技重点实验室，西安 710065)

摘　要：现有目标检测算法主要以图像中的大目标作为研究对象，针对小目标的研究比较少且存在检测精确度低、无法满足实时性要求的问题，基于此，提出一种基于深度学习目标检测框架 PVANet 的实时小目标检测方法。首先，构建一个专用于小目标检测的基准数据集，它包含的目标在一幅图像中的占比非常小且存在截断、遮挡等干扰，可以更好地评估小目标检测方法的优劣；其次，结合区域建议网络(RPN)提出一种生成高质量小目标候选框的方法以提高算法的检测精确度和速度；选用两种新的学习率策略“step”和“inv”以改善模型性能，进一步提升检测精确度。在构建的小目标数据集上，相比原 PVANet 算法平均检测精确度提高了 10.67%，速度提升了约 30%。实验结果表明，提出的方法是一个有效的小目标检测算法，达到了实时检测的效果。

关键词：小目标检测；小目标数据集；PVANet 算法；区域建议网络；学习率策略

中图分类号：TP391.41　　doi: 10.19734/j.issn.1001-3695.2018.06.0577

Real-time small object detection method based on improved pvanet

Duan Binghuan†, Wen Pengcheng, Li Peng

(*Aviation Key Laboratory of Science & Technology on Airborne & Missile-borne Computer, AVIC Xi'an Aeronautical Computing Technique Research Institute, Xi'an* 710065, *China*)

Abstract: Existing object detection algorithms are mainly aimed at detecting big objects in an image. Research on small object detection is still too scarce and there are problems with low detection accuracy and failure to meet the real-time requirement. This paper proposed a real-time small object detection method based on deep learning framework PVANet. Firstly, it built a benchmark dataset especially for small object detection problem. The dataset consisted of small objects covering a very small part of an image and also contained some interferences such as truncation and overlap. Secondly, combining with the Region Proposal Network (RPN) , it designed a strategy to generate high-quality candidate proposals for small objects to raise the detection accuracy and speed. Finally, it adopted two new learning rate policies "step" and "inv" to further enhance the detection accuracy. The proposed method achieved the mAP(mean average precision) by 10.67% and speed by 30% improvement over the original PVANet algorithm. Experimental results shows that this method is effective on small object detection and can run in real time.

Key words: small object detection; small object dataset; pvanet algorithm; region proposal network(rpn); learning rate policy

0　引言

在基于航拍的资源勘探、地震火灾救援等实际应用中，由于拍摄距离远，以致拍摄到的目标成像比较小，而复杂的背景信息又会对检测造成干扰，如何实时检测出这样的小目标成为研究的难点与热点问题。

近年来，各种目标检测算法如 Faster R-CNN[1]、SSD[2]、YOLOv2[3]等在计算机视觉领域取得了显著的成果，表现为在 PASCAL VOC[4]等通用数据集上的检测性能不断提高。这些通用数据集的图像中包含的目标通常在整张图中占有比较大的比例，而文献[5]评估发现，上述目标检测算法对图像中的小目标测试精确度较差，无法满足小目标检测应用的需求。

有一些学者针对小目标检测问题已做了相关的研究。Chen 等人[6]将首次将上下文信息与 R-CNN 算法[7]相结合进行小目标检测，与传统目标检测算法相比提高了测试精确度，但仍存在效率低、占存储空间大的问题。随后，一些研究者将改进的 R-CNN 算法——Fast R-CNN[8]和 Faster R-CNN 算法用于小目标检测以提升测试精确度和速度：文献[9，10]借助 Fast R-CNN 的上下文信息对小物体进行检测以提升检测性能；文献[11]利用 Faster R-CNN 检测行人，分析行人检测的误差主要来源于低分辨率的特征图和背景干扰，通过对区域建议网络(region proposal network，RPN)[1]进行修改来提升检测精确度；文献[12]利用 Faster R-CNN 来检测公司的标志这种小目标，在文献[11]的基础上进一步分析目标的大小及不同层级的特征图对检测效果的影响。此外，也有学者针对某一类小目标设计新的网络结构进行目标检测，文献[13]中提出一种端到端的卷积神经网络来检测小的交通标志，在精确度和速度方面都优于 Fast R-CNN 算法。虽然这些研究在小目标检测问题上取得了不少成果并且提供了很多新颖的思路，但它们研究的小目标在图像中仍然占有比较大的比例，实时性处理方面也达不到要求。

由此本文主要研究实时小目标检测问题。对于小目标，

收稿日期：2018-06-22；**修回日期：**2018-08-03　**基金项目：**航空科学基金资助项目（2015ZC31005，2017ZC31008）

作者简介：段秉环（1992-），女（通信作者），甘肃白银人，硕士研究生，主要研究方向为深度学习、计算机视觉（dbh_2010@126.com）；文鹏程（1981-），男，湖南长沙人，高级工程师，博士，主要研究方向为智能计算、计算机视觉、人机交互；李鹏（1977-），男，陕西西安人，研究员，主要研究方向为计算机体系结构、智能计算.

本文不将它们限制为现实世界中尺寸较小的物体，而是指广义上的小目标，即那些在一幅图像中占据比例很小的物体。对小目标进行检测主要需考虑以下难点。首先，与整张图像相比，需检测的目标占的比例很小，背景信息会对检测造成很大干扰，会大大增加精准定位小目标的困难；其次，与较大的目标相比，小目标的像素数更少，因此能提取到的有效的特征信息就会更少；另外，一幅图像中小目标数量往往比较多而且在实际应用中小目标之间经常互相重叠，这会进一步增加检测的难度。

PVANet[14]是一种可用于实时目标检测的深但轻量化的卷积神经网络，它用特征提取网络生成特征向量图，再基于 Faster R-CNN 算法中提出的 RPN 生成高质量的目标候选框(region proposal)用于后续的目标检测和定位。在通用数据集如 PASCAL VOC 上的测试结果表明，PVANet 算法的性能要优于 Faster R-CNN、SSD、YOLOv2 等算法。特别地，PVANet 的特征提取层中的卷积核较小，因此可以尽可能多地保留低层特征，这对小目标检测是有利的。综上，本文通过改进 PVANet 算法以提高小目标的检测性能，主要贡献如下：

a)构建专用于小目标检测的基准数据集。相比其他小目标检测研究中采用的数据集，该数据集中目标在图像中占更小的比例，且截断、遮挡等不完整的目标信息会增加检测难度，可以训练出性能更稳定的小目标检测模型。

b)针对原 PVANet 算法对小目标定位差的问题，提出一种生成高质量的小目标候选框的方法，提升了检测的精确度和速度；另外根据训练模型的特点选用两种新的学习率策略进一步改善模型性能。

1　构建小目标数据集

数据集是分析和评估基于深度学习的网络模型好坏的关键因素。Neovision2 Tower 数据集[15]是美国国防高级研究计划局(Defense Advanced Research Projects Agency, DARPA)构建的用于目标检测和实时跟踪的视频图像数据集，由于拍摄距离远，数据集中的目标比较小，而且拍摄场景中目标多且杂乱，存在着可变光照和遮挡干扰，这些都使其成为具有挑战性的目标检测数据集。因此本文选取 Neovision2 Tower 数据集来构建小目标数据集。Neovision2 Tower 数据集包含 100 个视频片段，每个视频片段已截取成 900 张高分辨率 1920×1080 像素的 PNG 图片集，高清图像可以尽可能多地保留小目标信息。本文主要从以下几方面构建小目标数据集，为提高通用性，格式参照 PASCAL VOC。

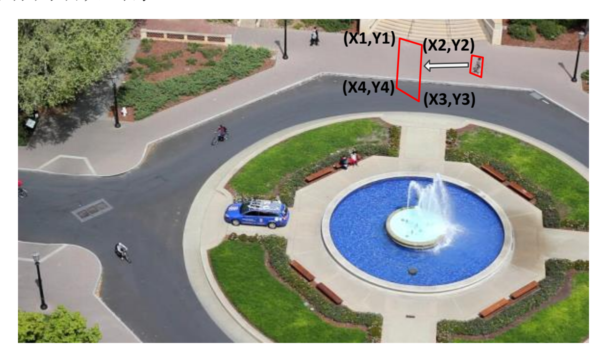
a)对图像降维，使其大小为 960×544 像素，并将其压缩为.jpg 格式，以使数据格式与 PASCAL VOC 等常用数据集相同，这对加速网络模型的训练过程是必要的。

b)修改目标的标注框。原数据集中目标的标注信息由 4 个边界坐标—(X1,Y1)、(X2,Y2)、(X3,Y3)、(X4,Y4)组成。虽然这 4 个坐标构成的边界框尽可能地贴近了目标，但它不是矩形，不符合通用数据集规范。为此修改标注框为矩形框，以左上角坐标(Xmin,Ymin)和右下角坐标(Xmax,Ymax)来确定它，如图 1 所示,新坐标可以表示为：

$$\begin{aligned} Xmin &= \min(X1, X2, X3, X4) \\ Ymin &= \min(Y1, Y2, Y3, Y4) \\ Xmax &= \max(X1, X2, X3, X4) \\ Ymax &= \max(Y1, Y2, Y3, Y4) \end{aligned} \tag{1}$$

图 1(b)中还用椭圆形标记举例了构建的数据集中目标存在截断和遮挡等干扰的情况。

最后，将已处理后的 Tower 数据集划分为两个子集：训练集和测试集，每个子集分别对应 50 个图片集。然后从训练集中随机选取 10 个图片集作为验证集，剩余 40 个图片集供训练用。Neovision2 Tower 数据集包含人、自行车、小汽车、卡车、公交车共五类目标，本文选取人和自行车这两种小目标作为研究对象。



(a) Neovision2 Tower 数据集的目标标注举例



(b) 本文的小目标数据集的目标标注举例

图 1　两种数据集的目标标注形式示意图

Fig. 1　Diagram of the bounding boxes of the object in tow datasets

构建的小目标数据集包含两种拍摄角度的图像：第一种图像是由具有固定倾斜角度的相机俯拍获得，另一种是第一拍摄视角旋转 90 度而成。采用这两种视角的图像可以增加训练数据的多样性，并使检测模型具有更好的泛化性能。在本文构建的小目标数据集中，人的平均大小为 17×24 个像素，在 960×544 像素的整幅图片中占比例约为 0.078%；自行车的平均大小为 40×38 像素，在一幅图片中占比例约为 0.291%。本文提出的小目标数据集中目标平均占整幅图像比例为 0.184%，与 PASCAL VOC 通用数据集和文献[6]中提出的小目标数据集相比，目标占比更小，如表 1 所示。

表 1　本文构建的小目标数据集与其他数据集的对比

Table 1　Comparison of small object dataset in this paper with others

数据集	PASCAL VOC 数据集	文献[7]中的小目标集	本文的小目标集 人	本文的小目标集 自行车	本文的小目标集 平均
目标平均占整幅图像比例(%)	16.177	0.321	0.078	0.291	0.184

文献[6]中提出的小目标数据集是比较优秀的构建数据集的范例，被不少研究者采用进行小目标检测。本文设计的小目标数据集与文献[6]中的相比至少有以下两方面的优点：首先，本文构建的小目标数据集中的目标更小，而且还有背景对其的被动遮挡和目标之间的主动遮挡，另外由于数据集来源于视频图像，目标是在不断移动变换位置的，因此在图像的边界处会被截断，这些都增加了小目标检测的困难，更能评估模型在小目标检测上的优劣；其次，本文设计的数据集中采用视频图像，由于图像之间存在着时序信息、互相关联，可以对目标的形态连续采样，有利于训练出更具有鲁棒性的检测模型。

2　改进 PVANet 用于小目标检测

2.1　PVANet 的基本原理

PVANet 是一种轻量级的目标检测算法，它主要分两阶段实现。首先，特征提取网络输出特征图到 RPN 生成目标候选框。其次，上阶段生成的目标候选框及特征图经过池化层和全连接层后送入分类层以确定目标的类型以及同时送入边界框回归层进一步调整目标边框的位置。PVANet 整体框架结构如图 2 所示。

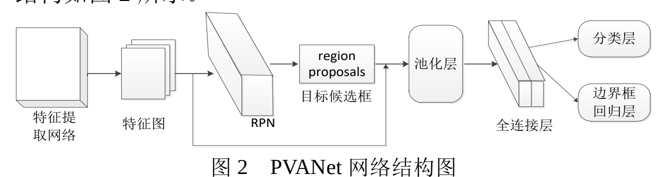


图 2　PVANet 网络结构图

Fig. 2　Architecture of pvanet

PVANet 的贡献主要在于提出了一个高效的特征提取网络，基于层数多但通道少的设计原则，采用 C.ReLU[16]、Incepion[17]、HyperNet[18]和残差连接[19]等技术来生成特征图，实现了加速模型性能而不会降低检测精确度这一目标。PVANet 的特征提取网络如图 3 所示。

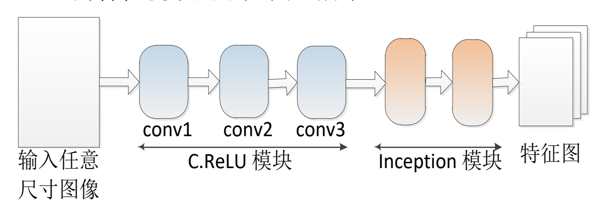


图 3　PVANet 的特征提取网络示意图

Fig. 3　Feature extraction network of pvanet

PVANet 的特征提取网络的前几层由 C.ReLU 模块构成。研究发现，卷积神经网络(CNN)中前几层的卷积核之间存在着负相关。利用此特点，C.ReLU 简单地连接每个卷积核的输出值和它的取反值，再缩放或移位，然后进行 ReLU 计算，这使得每个通道的斜率和激活阈值不同于其相反的通道，而且使输出通道的数量减少了一半即不需要存储其相反通道的参数，同时没有失去准确性。C.ReLU 模块的采用是 PVANet 能实现轻量化的重要原因。另外，Inception 模块被用于剩余的特征提取网络。作为可以同时捕获图像中的小目标和大目标的最具有成本效益的构件之一，Inception 模块可以为不同大小的感受野生成激活值。特别地，Inception 模块中的 1×1 卷积核有利于定位小目标候选框并能更精准地捕获小目标。

总之，对于实时小目标检测，PVANet 至少在以下三个方面优于其他算法：首先，它采用 C.ReLU 模块来减少计算量以提高检测速度。此外，它还采用 RPN 网络来生成高质量的目标候选框。特别地，Inception 模块的选用可以使它尽可能多地存储低层网络的必要信息，这对小目标检测是有利的。

2.2　针对小目标检测对 PVANet 的改进

文献[6]指出，小目标检测的挑战主要来自于目标候选框的生成。因此，本文致力于生成高质量的小目标候选框，主要通过在 RPN 网络中设置合适的锚框来实现。另外，与其他的超参数相比，学习率是影响目标检测性能的最重要的参数之一，并且以更复杂的方式控制着模型的有效容量。当学习率最优时，模型的有效容量最大。基于此，本文将比较不同的学习率策略并选择最优的策略微调模型以提升小目标检测性能。

2.2.1 生成小目标候选框

SeletiveSearch[20]和 EdgeBoxes[21]是目标检测中常用的生成目标候选框的方法，已经在通用数据集如 PASCAL VOC 上取得了比较好的结果。不过，SeletiveSearch 生成目标候选框的速度很慢，在 CPU 上处理一幅图像大约需要 2 秒的时间。虽然 EdgeBoxes 在生成目标候选框的质量和处理速度之间达到了很好的平衡，但处理一幅图像仍然需要 0.2s[1]。与整条目标检测线相比，上两种方法生成目标候选框消耗的时间太多，因此它们都不能满足实时性要求。此外，SeletiveSearch 和 EdgeBoxes 在生成大目标的候选框时表现良好，但在生成小目标候选框时效果较差，测试发现是因为这两种方法对目标的重要特征比如轮廓和独特的颜色等表现敏感，而小目标通常本身包含很少的信息，因此这两种方法无法生成高质量的小目标候选框。

RPN 已被证明是当前最优的生成目标候选框的方法，它大大缩短了目标候选框的生成时间。它通过在特征提取网络生成的特征图上应用 3×3 滑动窗口(sliding window)和锚框(anchor box)输出 512 维的特征，然后将其输入到后续的两个子全连接层—分类层和边界框回归层。分类层预测目标候选框分别是前景和背景的概率，边界框回归层输出目标候选框的 4 个位置坐标。在 PVANet 的最早版本中，滑动窗口的每个位置处产生 25 个锚框，由 5 个不同的尺度(96、192、288、512、800)和 5 个不同的纵横比(0.5、0.667、1.0、1.5、2.0)确定。本文构建的小目标数据集中，人和自行车的平均大小分别为 17×24 像素、40×38 像素。显然 RPN 的原始尺度对于本文的小目标来说太大了，将其直接用于检测小目标时精确度较差，所以需要缩小锚框以适应小目标的尺寸。最新版本的 PVANet 使用了 6 种尺度(32、48、80、144、256、512)和 7 个纵横比(0.333、0.5、0.667、1.0、1.5、2.0、3.0)构成 42 个锚框[14]。新版本增加了锚框的数量以扩大目标检测的范围，并且与其最初版本比较，在 PASCAL VOC 上测试的平均精确度提高了近 3%，但对比发现这些锚框的尺寸变化范围太大且它的最小的尺寸都比本文构建的小目标的平均尺寸大。

本文构建的小目标数据集中目标的尺寸并没有非常大的变化，特别是小目标人和自行车的尺度差不超过 20 像素，所以本文减少了锚框尺寸的数量和大小，但因为人和自行车的边界框形状主要是矩形，因此同时尽可能多地保持锚框纵横比，以更精准地定位小目标。

基于此，本文为滑动窗口的每个位置选择 24 个锚框，包含 4 种尺寸(16、24、32、64)和 6 种纵横比(0.333、0.5、0.667、1、1.5、2)，RPN 结构如图 4 所示。

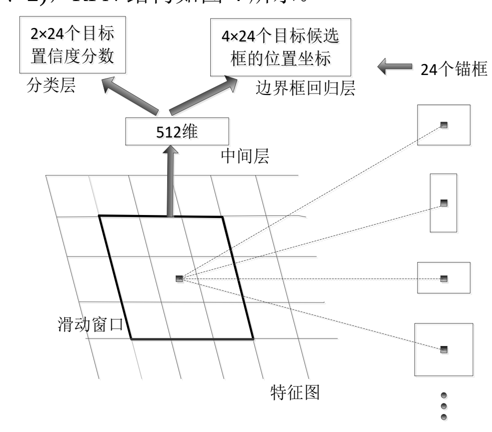


图 4　本文对 RPN 网络的实现

Fig. 4　Implementation of RPN in this paper

本文提出的方法与原 PVANet 方法的检测效果如图 5 所示。



PVANet 方法检测图示例



本文方法检测图示例

(a)俯拍视角、暗光线下两种算法检测效果图



PVANet 方法检测图示例



本文方法检测图示例

(b)旋转 90 度视角、亮光线下两种算法检测效果图

图 5 两种算法检测效果对比示例图

Fig. 5 Comparison of the detecting results of two algorithms

图 6 中(a)和(b)分别是两种视角和不同的光照条件下两种算法的检测效果对比示例图，其中目标框上显示的是目标的类别和置信度分数。比较发现原 PVANet 算法检测小目标时会出现很多将背景误判成目标的误检框，特别在目标之间互相存在遮挡的情况下，如(a)左边的子图所示；另外截断的目标由于信息量少容易被漏检，如(b)左边的子图所示。改进后的 PVANet 算法由于生成了更高质量的小目标候选框，因此对小目标的定位更准确，可以有效地抵抗目标间互相遮挡的干扰，误检框比较少，而且可以正确检测出被截断的目标，如(a)右边的子图和(b)右边的子图所示。

2.2.2 选用新的学习率策略

学习率是深度学习中一个非常重要的超参数，它可以指导人们通过损失函数的梯度来调整网络的权重。一般而言更好的学习率策略意味着可以在更短的时间内训练出更优的网络模型，因此调整学习率是通过训练过程来提升模型性能的重要手段之一。

PVANet 基于“plateau”策略[14]来动态控制学习速率，该策略监控损失函数变动的平均值，发现若在某段迭代周期中其改善低于某一个阈值，则确定损失函数的变化此时处于一个“高原”，学习率则降低一个常数因子。然而，本文首先采用“plateau”学习率策略来训练模型，设定迭代次数为 100000 次，发现学习率一直保持在初始值 0.001 不变。学习率不变的主要原因在于训练过程中目标区域相比背景区域非常小，这导致了负样本空间大，模型本身会收敛的比较慢，因此直接采取通过评估损失函数的动态均值来改变学习率的“plateau”策略训练模型很难变化学习率。因此需采取其他的学习率策略来改变学习率以加速模型收敛。

观察损失函数曲线发现它在 50000 次迭代后趋于平坦，测试发现 50000 次迭代后检测精确度提高地很缓慢，为此猜测训练到 50000 次时，损失函数的梯度已接近“高原”状态，此时训练损失已很难得到改善。为了帮助损失函数尽快走出“高原”状态，采用“step”学习率策略[22]，其计算公式定义为

$$\text{learningRate}=\text{base_}lr\times\text{gamma}^{(\text{floor}(\text{iter}|\text{stepsize}))} \tag{2}$$

其中：learningRate 指学习率，base_lr 指初始学习率，gamma 和 stepsize 是参数，iter 指迭代次数，当迭代次数达到 stepsize 的整数倍时学习率开始降低。本文将初始学习率设置为 0.001，并在 50000 次迭代后将其降至 0.0001，迭代 100000 次后测试发现比采用“plateau”学习率策略检测精确度提升了约 0.45%。

尽管低的学习率可以保证人们不会错过任何最小值，但这也意味着不得不花费更多时间使模型收敛，特别是当损失函数陷入“高原状态”时。文献[23]指出减少损失的难度主要来自鞍点而不是误差曲面上的局部最小点。考虑到这个因素，本文尝试另一种学习率策略—“inv”[22]，它动态地改变学习率来加速损失函数的收敛，而不是像“step”策略那样均匀地改变。将初始学习率设置为 0.001，参数 gamma 和 power 的值分别设为 0.0001 和 0.75。“inv”学习率公式定义为

$$\text{learningRate}=\text{base_}lr\times(1+\text{gamma}\times\text{iter})^{(-\text{power})} \tag{3}$$

其中：learningRate 指学习率，base_lr 指初始学习率，iter 指迭代次数，gamma 和 power 是参数。设置训练次数为 100000 次，学习率在迭代的第 50000 次“高原”处时动态减小到 0.00026，在 100000 次迭代后为 0.00017。实验结果表明采用“inv”学习率策略比用“plateau”和“step”策略能得到性能更好的网络模型，更适合本文的小目标检测场景。

2.3 实验结果及分析

本文使用 Quadro K6000 GPU 进行相应的实验，为评估本文的算法在小目标检测问题上的有效性，采用平均准确率(average precision,AP) 和所有类别的平均准确率(mean

average precision,mAP)作为衡量模型性能的评价指标。AP 是评价单个类别检测精确度的最直观的指标，mAP 是所有类别 AP 的均值，可以评价模型的综合性能。表 2 列出了本文的方法与目前主流的目标检测算法 Faster R-CNN、SSD、YOLOv2、PVANet 在构建的小目标数据集上的测试精确度和运行时间(frames per second, FPS）的对比情况。

表 2　几种算法的测试精确度和运行时间对比

Table 2　Comparison of test accuracy and runtime of several algorithms

算法(学习率策略)	AP(人)/%	AP(自行车)/%	mAP/%	FPS
Faster R-CNN(step)	35.17	82.81	58.99	1.6
SSD300 (step)	32.10	76.37	54.24	15
SSD500(step)	34.89	78.22	56.56	6
YOLOv2 416×416(step)	29.23	74.34	51.79	31
YOLOv2 544×544(step)	31.36	75.86	53.61	22
PVANet 的最初版本(plateau)	39.20	83.65	61.42	7
PVANet 的最新版本(plateau)	48.00	86.52	67.26	
本文的方法(plateau)	53.64	88.26	**70.95**	10
本文的方法(step)	54.18	88.61	**71.40**	
本文的方法(inv)	55.46	88.71	**72.09**	

1）采用不同的方法检测效果分析

现有基于深度学习的目标检测算法可大致分为两类，一类算法先产生目标候选区域，再进行目标分类和目标边界框预测，以 Faster R-CNN、PVANet 等为代表，这类算法可以较好地定位目标，但检测速度较慢；第二类算法是直接预测目标置信度分数和目标边界框的端到端的目标检测框架，以 YOLO、SSD 算法为代表，其优点是网络结构简单，测试速度快，但不能很好地确定目标位置，尤其对相邻或相近的目标测试精确度差。由表 2 可看出，PVANet、Faster R-CNN 算法在小目标上的检测精确度比 YOLOv2、SSD 的各个版本都高；YOLOv2 在表 2 列出的所有算法中检测速度最快，但测试精确度最低；SSD 结合了 YOLO 和 Faster R-CNN 算法，用多尺度的思想来预测目标边界框，提高了测试精确度。本文提出的生成小目标候选框的方法由于充分考虑了小目标检测的特点，能较多地提升算法性能。对比表 2 中各种算法可发现，本文的方法在检测速度方面逊于 YOLOv2 算法，但在保证基本实现实时检测的情况下，测试精确度明显优于其他方法，综合考虑本文提出的方法是一个有效的小目标检测算法。

2）采用不同策略训练网络检测效果分析

学习率对模型收敛到局部极小值即达到最高的精度的影响是很大的。本文的方法首先沿用 PVANet 中的“plateau”学习率策略来训练网络，由表 2 可看到，平均测试精确度达到了 70.95%，相比较原 PVANet 算法，检测精确度提升了 9.53%。但在小目标检测问题中，目标区域占比太小，负样本空间大，模型收敛速度慢；且 PVANet 模型只卷积层和全连接层就有 94 层，网络深且窄，这决定了在训练过程中模型的损失函数很容易震荡。“plateau”学习率策略通过监控损失函数的变动值来改变学习率，当变动值在某一段时间内小于设定的阈值时降低学习率；而当损失函数开始震荡且一段时间内震荡的变动值又大于监控的阈值时，学习率不会变化，损失函数也不会再收敛。为进一步提升检测精确度，本文分别采用均匀变化学习率的策略“step”和动态变化学习率的策略“inv”来训练网络。采用这两种学习率策略即便当损失函数陷入振荡期，随着迭代次数的增加学习率仍然会减小，模型可进一步收敛，由表 2 可看出采用这两种学习率策略比用“plateau”学习率策略检测精确度分别提高了 0.45%和 1.14%。此外，“step”学习率策略需要手动设置降低学习率的迭代次数间隔，而“inv”学习率策略使得学习率每一次迭代时都减小，每次减小的是一个很小的数，省去了手动设置学习率变动的迭代间隔可能造成的不当。表 2 实验结果表明采用“inv”动态学习率策略能训练出最优的网络模型，比用“step”学习率策略检测精确度提高了 0.69%。本文的方法用“inv”学习率策略在构建的小目标数据集上达到了 72.09%的平均测试精确度，比原 PVANet 算法检测性能提升了 10.67%，可见学习速率的动态变化对于更快跨越训练过程中误差曲面的鞍点并提高检测精确度起着重要作用。

3）GPU 性能对运行时间的影响

在时间性能上，文献[14]指出 PVANet 在 NVIDIA Titan X GPU 上对 1056×640 像素的图片测试速度可以达到 21.7FPS。本文采用 Quadro K6000 GPU 进行测试，对于 960×544 像素的图片，用原 PVANet 算法测试速度为 7FPS。存在此差异的主要原因在于 NVIDIA Titan X GPU 的计算能力为 6.1，而 Quadro K6000 GPU 的计算能力为 3.5[24]，基本上是前者的一半；另外本文的测试集中的每张图片包含的目标数量更多，一定程度上也耗费了检测时间。本文通过生成高质量的小目标候选框，在 Quadro K6000 GPU 上的测试速度为 10FPS，比原 PVANet 算法在速度上提升了 30%。

3　结束语

本文主要改进了目前目标检测领域性能很优的 PVANet 算法使其适用于实时小目标检测场景。通过将生成高质量小目标候选框的方法与 RPN 网络相结合，并选用合适的学习率策略，有效地解决了小目标检测中因目标尺寸很小及目标存在截断和遮挡干扰以致定位困难的难点问题。实验证明，本文的方法在小目标检测上具有很好的鲁棒性并且在 Quadro K6000 GPU 上的测试速度比原 PVANet 算法提高了 41ms/image，达到了实时检测的效果。由于本文的数据集是视频图像，下一步工作可以着眼于进一步提升检测速度使其可用于视频检测。

参考文献：

[1] Ren Shaoqing, He Kaiming, Girshick R, *et al.* Faster R-CNN: towards real-time object detection with region proposal networks [C]// Proc of International Conference on Neural Information Processing Systems. Cambridge: MIT Press, 2015: 91-99.

[2] Liu Wei, Anguelov D, Erhan D, *et al.* SSD: single shot multibox detector [C]//Proc of European Conference on Computer Vision. New York: Springer, 2016: 21-37.

[3] Redmon J, Farhadi A. YOLO9000: better, faster, stronger [C]//Proc of IEEE Conference on Computer Vision and Pattern Recognition. Piscataway, NJ: IEEE Press, 2017: 6517-6525.

[4] Everingham M, Van Gool L, Williams C K I, *et al.* The pascal visual object classes (voc) challenge [J]. International Journal of Computer Vision, 2010, 88 (2): 303-338.

[5] Pham P, Nguyen D, Do T, *et al.* Evaluation of deep models for real-time small object detection [C]//Proc of International Conference on Neural Information Processing. New York: Springer, 2017: 516-526.

[6] Chen Chenyi, Liu Mingyu, Tuzel O, *et al.* R-CNN for small object detection [C]//Proc of Asian Conference on Computer Vision. New York: Springer, 2016: 214-230.

[7] Girshick R, Donahue J, Darrell T, *et al.* Rich feature hierarchies for

accurate object detection and semantic segmentation [C]// Proc of IEEE Conference on Computer Vision and Pattern Recognition. Piscataway, NJ: IEEE Press, 2014: 580-587.

[8] Girshick R. Fast R-CNN [C]//Proc of IEEE Conference on Computer Vision and Pattern Recognition. Piscataway, NJ: IEEE Press, 2015: 1440-1448.

[9] Bell S, Lawrence Zitnick C, Bala K, *et al.* Inside-outside net: detecting objects in context with skip pooling and recurrent neural networks [C]// Proc of IEEE Conference on Computer Vision and Pattern Recognition. Piscataway, NJ: IEEE Press, 2016: 2874-2883.

[10] Eggert C, Winschel A, Zecha D, *et al.* Saliency-guided selective magnification for company logo detection [C]//Proc of the 23rd International Conference on Pattern Recognition. Piscataway, NJ: IEEE Press, 2016: 651-656.

[11] Zhang Liliang, Lin Liang, Liang Xiaodan, *et al.* Is Faster R-CNN doing well for pedestrian detection? [C]//Proc of European Conference on Computer Vision. New York: Springer, 2016: 443-457.

[12] Eggert C, Brehm S, Winschel A, *et al.* A closer look: small object detection in Faster R-CNN [C]//Proc of IEEE International Conference on Multimedia and Expo. Piscataway, NJ: IEEE Press, 2017: 421-426.

[13] Zhu Zhe, Liang Dun, Zhang Songhai, *et al.* Traffic-sign detection and classification in the wild [C]//Proc of IEEE Conference on Computer Vision and Pattern Recognition. Piscataway, NJ: IEEE Press, 2016: 2110-2118.

[14] Sanghoon H, Roh B, Kim K H, *et al.* PVANET: deep but lightweight neural networks for real-time object detection [C]//Proc of International Conference on Neural Information Processing Systems.Canbridege: MIT Press, 2016: 1608-1614.

[15] Khosla D, Chen Yang, Kim K. A neuromorphic system for video object recognition [J]. Frontiers in Computational Neuroscience, 2014, 8(8): 1-12.

[16] Shang Wenling, Sohn K, Almeida D, *et al.* Understanding and improving convolutional neural networks via concatenated rectified linear units [C]//Proc of International Conference on Machine Learning. New York: ACM Press, 2016: 2217-2225.

[17] Szegedy C, Liu Wei, Jia Yangqing, *et al.* Going deeper with convolutions [C]// Proc of IEEE Conference on Computer Vision and Pattern Recognition. Piscataway, NJ: IEEE Press, 2015: 1-9.

[18] Kong Tao, Yao Anbang, Chen Yurong, *et al*. Hypernet: towards accurate region proposal generation and joint object detection [C]// Proc of IEEE Conference on Computer Vision and Pattern Recognition. Piscataway, NJ: IEEE Press, 2016: 845-853.

[19] He Kaiming, Zhang Xiangyu, Ren Shaoqing, *et al.* Deep residual learning for image recognition [C]//Proc of the IEEE Conference on Computer Vision and Pattern Recognition. Piscataway, NJ: IEEE Press, 2016: 770-778.

[20] Uijlings J R, Sande K E, Gevers T, *et al.* Selective search for object recognition [J]. International Journal of Computer Vision, 2013, 104(2): 154-171.

[21] Zitnick C L, Dollár P. Edge boxes: locating object proposals from edges [C]//Proc of European Conference on Computer Vision. New York: Springer, 2014: 391-405.

[22] Jia Yangqing, Shelhamer E, Donahue J, *et al.* Caffe: convolutional architecture for fast feature embedding [C]//Proc of the 22nd ACM international conference on Multimedia. New York: ACM Press, 2014: 675-678.

[23] Smith L N. Cyclical learning rates for training neural networks [C]// Proc of IEEE Winter Conference on Applications of Computer Vision. Piscataway, NJ: IEEE Press, 2017: 464-472.

[24] CUDA GPUs [EB/OL]. [2018-05-03]. https://developer. nvidia. com/cuda-gpus.